

## **REMARKS**

In the Official Action mailed on **02 January 2009**, the Examiner reviewed claims 1-28. Examiner rejected claims 1-28 under 35 U.S.C. § 102(b) based on Hanson et al. (U.S. Patent No. 6,148,346, hereinafter “Hanson”).

### **Rejections under 35 U.S.C. § 102(b)**

Examiner rejected Independent claims 1, 11, and 20 as being anticipated by Hanson. Applicant respectfully disagrees with this rejection. Hanson does not disclose a universal contextual interface that does **not** have a priori knowledge of the devices’ **file system domain protocol** or the devices’ **printer domain protocol**, where the devices’ file system domain protocol comprises **Network File System** (NSF) or **Common Internet File System** (CIFS), and where the devices’ printer domain protocol comprises **Internet Printing Protocol** (IPP) or **Line Printer Daemon**.

Hanson discloses a **device driver** with “an operating system (OS) independent device driver portion and an OS specific device driver portion” (Hanson, C3:L27-30). As is well-known in the art, a device driver **converts** data from a file or an application into a form that can then be used by a hardware device. For example, a printer driver converts data to be printed to the form specific to the printer. Thus a device driver acts as an abstraction layer between a hardware device and applications and operating systems that use the device. Hanson, C4:L21-57 discloses exactly such a device driver. For example,

...if the peripheral device is a printer, the peripheral specific data objects 54 include information about the printer’s paper trays, the printer’s formatting requirements, etc. (Hanson, C4:L52-55).

Elsewhere, Hanson expressly states that the device driver **formats** the data for a peripheral device: “...the document must be correctly formatted by a device driver specific to the peripheral device” (Hanson, C3:L34-36). In short, the

function of the device driver in Hanson (and as well-known in the art) is to **convert** data.

Moreover, in the Hanson system, communication to a device **requires** at least some type of file system or communication protocol such as a file system domain protocol (e.g., NFS or CIFS) or a printer domain protocol (e.g., IPP or LPR). Hanson discloses a File Transfer Protocol (FTP) in Hanson, C3:L32-35), but is silent on other such protocols.

In contrast, embodiments of the present invention involve a universal contextual interface that does **not** have a priori knowledge of the devices' **file system domain protocol** or the devices' **printer domain protocol**, where the devices' file system domain protocol comprises **Network File System** (NFS) or **Common Internet File System** (CIFS), and where the devices' printer domain protocol comprises **Internet Printing Protocol** (IPP) or **Line Printer Daemon** (instant application, pars. [0001] and [0003]-[0004]). NFS allows users to access files across a network and treat them as if they resided in a local file directory. CIFS allows programs to make requests for files and services on remote computers on the Internet. Furthermore, IPP includes functions such as allowing the user to submit a print job, finding out about the printer's capabilities, finding out about a print job's status, and cancelling a print job. Similarly, LPD is a set of programs that provide printer spooling and network print server functionality for Unix-like systems, including assigning a job to a queue, displaying jobs assigned to a queue, removing a job from a queue, and controlling a queue.

Neither NFS, CIFS, IPP, nor LPD **converts** data between the operating system or an application and a device. That task is the job of a **device driver**. Note that an operating system might have a device driver that tells it how to convert a document into a specific printer's language but it might not be able to communicate with the printer to submit a job (i.e., it might not know IPP or LPD). Thus, NFS, CIFS, IPP, and LPD are **not** the same as device drivers.

Note that Hanson does not disclose the following **negative** limitation: a universal contextual interface that does **not** have a priori knowledge of the

devices' **file system domain protocol** or the devices' **printer domain protocol**, where the devices' file system domain protocol comprises **Network File System** (NSF) or **Common Internet File System** (CIFS), and where the devices' printer domain protocol comprises **Internet Printing Protocol** (IPP) or **Line Printer Daemon**.

Accordingly, Applicant has amended independent claims 1, 11, and 20 to clarify that embodiments of the present invention involve the aforementioned features. These amendments find support in instant application, pars. [0001] and [0003]-[0004]. No new matter has been added.

Hence, Applicant respectfully submits that independent claims 1, 11, and 20 as presently amended are in condition for allowance. Applicant also submits that claims 2-10, which depend upon claim 1, claims 12-19, which depend upon claim 11, and claims 21-28, which depend upon claim 20, are for the same reasons in condition for allowance and for reasons of the unique combinations recited in such claims.

## **CONCLUSION**

It is submitted that the present application is presently in form for allowance. Such action is respectfully requested.

Respectfully submitted,

By /Shun Yao/  
Shun Yao  
Registration No. 59,242

Date: 1 April 2009

Shun Yao  
PARK, VAUGHAN & FLEMING LLP  
2820 Fifth Street  
Davis, CA 95618-7759  
Tel: (530) 759-1667  
Fax: (530) 759-1665  
Email: shun@parklegal.com